SUBSTITUTE SPECIFICATION

METHOD FOR MIN-CUT AND RATIO MIN-CUT PARTITIONING

Field of the Invention

The present invention relates to a method for min-cut and ratio min-cut partitioning, and more specifically, to an optimal and intuitive heuristic optimal method for min-cut partitioning.

Background of the Invention

The large integration of semiconductor ICs has been accomplished by a reduction in individual device size. With this reduction of device size, many challenges arise in the manufacture of integrated circuits. Integrated circuits typically include a great number of electronic components fabricated in multi-layers with several different materials on a wafer. IC design includes the technique of circuit design to create a schematic design for a desired circuit.

An actual device is produced to perform the function described in the schematic design. The transformation from the circuit description into a geometric description is referred to as a layout. A layout consists of a set of planar geometric shapes in several layers. The purpose of the layout procedure is to construct a device which reduces the area of the layout area and signal propagation delays between associated logic elements. Thus, the desired layout area and the signal propagation delays between elements are considered in the configuration of the element locations.

Circuit partitioning plays a key role in the field of chip design, multi-chip system and system-on-chip (SOC). Circuit partitioning is used to reduce VLSI chip area, reduce the component count and the number of interconnections in multiple FPGA implementations of large circuits or system,; facilitates efficient parallel simulation of circuits, facilitates design of tests for digital circuits and reduces timing delays, and facilitates the various combination of sub-system layouts.

The goal of circuit partitioning methods is to minimize the number of nodes that connect sub-circuits. In general, VLSI design uses computer-aided design tools to perform the partitioning. Up to now, the circuit simulation is executed using a computer system so that the circuit exhibits the desired performance.

Parallel simulation of a circuit is sufficient to test a circuit's design. Circuit simulation simulates systems by partitioning a target system into a plurality of sub-circuits for parallel simulation. In simulating systems, the partitioning method for the target circuit significantly affects the accuracy and the speed required for computations.

Some of the prior art may refer to Naveed. A. Sherwani, (Intel Corp.), Chapter 5: Partitioning, Algorithms for VLSI Physical Design Automation. 3rd Ed., Boston, MA: Kluwer Academic Publishers, 1999., and other references. However, the prior art is complicated and inefficient. For example, most partitioning methods for circuit netlists, like the Fiduccia-Mattheyses (FM) method, compute the gains of nodes using local netlist information (i.e., it only concerns the immediate improvement in the cutset).

What is needed is a method that involves the use of not only node information, but also edge information.

Other references

[1] S. B. Akers, “Clustering Techniques for VLSI,” in Proc. IEEE Int. Symp. on Circuits and Systems, 1982, pp. 472-476.

[2] C. J. Alpert, J.-H. Huang, and A. B. Kahng, “Multilevel circuit partitioning,” in Proc. Design Automation Conf, 1997, pp. 530-533.

[3] C. J. Alpert and S.-Z, Yao, “Spectral partitioning: The more eigen-vectors the better,” in Proc. Design Automation Conf, 1995, pp. 195-200.

[4] J. Cong et al., “Large scale circuit partitioning with loose/stable net removal and signal flow based clustering,” in Proc. IEEE/ACM mt. Conf Computer-A ided Design, Nov. 1997, pp. 441-446.

[5] J. Cong and S. K. Lim, “Multiway Partitioning with Pairwise Movement,” in Proc. Int. Conf. Computer-Aided Design, 1998, pp. 512-516.

[6] S. Dutt and W. Deng, "A probability-based approach to VLSI circuit partitioning," in Proc. iEEE/ACM Design Automation Conf, June 1996, Best-Paper Award, pp. 100-105.

[7] S. Dutt, "New faster Kernighan-Lin-type graph-partitioning algo-rithms," in Proc. IEEE/A CM In Conf Computer-Aided Design, Nov. 1993.

[8] C. M. Fiduccia and R. M. Mattheyses, "A linear-time heuristic for improving network partitions," in Proc. IEEE/ACM 19th Design Automation Conf, 1982, pp. 175-181.

[9] J. Garbers, H. J. Promel, and A. Steger, "Finding clusters in VLSI circuits," in Proc. IEEE/A CM Int. Conf Computer-A ided Design, 1990, pp. 520-523.

[10] M. R. Garey and D. S. Johnson, Computers and Intractability. San Francisco, CA: W. H. Freeman, pp. 209-210.

[11] L. Hagen and A. Kahng, "Fast spectral methods for ratio cut partitioning and clustering," in Proc. IEEE/ACM Int. Conf Computer-Aided Design, 1991, pp. 10-13.

[12] M. A. B. Jackson, A. Srinivasan, and E. S. Kuh, "A fast algorithm for performance driven placement," in Proc. IEEE/A CM mt. Conf. Computer-Aided Design, 1990, pp. 328-331.

[13] B. W. Kernighan and S. Lin, "An efficient heuristic procedure for partitioning graphs," Bell System Tech. Journal, vol. 49, pp. 291-307, Feb. 1970.

[14] D. B. Knuth, Sorting and Searching. Addison-Wesley, 1973.

[15] B. Krishnamurthy, "An improved min-cut algorithm for partitioning VLSI networks," IEEE Trans. Computers, vol. C-33, pp. 438-446, May 1984.

[16] Y. G. Saab, "A fast and robust network bisection algorithm," IEEE Trans. Computers, pp. 903-913, 1995.

[17] C. Sechen, VLSI Placement and Global Routing Using Simulated Annealing, B. V. Deventer, Ed. Amsterdam, Netherlands: Kiuwer.

[18] N. A. Sherwani, Algorithms for VLSI Physical Design Automation. 3rd Ed., Boston, MA: Kiuwer, 1999.

[19] Y. C. Wei and C. K. Cheng, "An improved two-way partitioning algorithm with stable performance," IEEE Trans. Computer-Aided Design, pp. 1502-1511, 1990.

[20] Y. C. Wei and C. K. Cheng, "Toward efficient hierarchical designs by ratio cut partitioning," in Proc. Int. Conf. Computer-Aided Design, 1989, pp. 298-301.

Summary of the Invention

An object of the present invention is to provide a method of min-cut and ratio min-cut partitioning. The present invention discloses a new method Edge-Node Interleave Sort for Leaching and Envelop (ENISLE), for min-cut partitioning. The method includes one step of mapping the circuit into a V-E plane and steps of sorting the V-E pairs contained on the V-E plane. ENISLE is a novel method rather than an improved or modified min-cut partitioning. The method not only uses node information, but also edge information. The (V, E) pairs approach uniform distribution on the V-E plane, to obtain the optimal solution.

ENISLE comprises mapping a circuit into a V-E plane to transform a circuit information onto a V-E plane. The V-E plane contains the information of node and edge information, wherein V indicates nodes that represent components of the circuit and wherein E indicates edges that represent the nets of the circuit. The next step is to determine whether (V, E) pair distribution on the V-E plane is uniform or not. If (V E) pair distribution is non-uniform, then realignment of the (V, E) pairs on the V-E plane according to the magnitude of each node or edge, thereby obtaining min-cut or/and ratio min-cut partitioning is accomplished by using the following steps:

(1) Performing a first sorting step from an edge view based on a first side of the V-E plane;

(2) Performing a second sorting step from an node view based on a second side of the V-E plane;

(3) Performing a third sorting from said edge view based on a third side of the V-E plane; and

(4) Performing a fourth sorting step from said node view based on a fourth side of the V-E plane.

Brief Description of the Drawings

FIG. 1 is a diagram of the goal of min-cut partitioning.

FIG. 2 is a diagram of the goal of ratio min-cut partitioning.

FIG. 3 shows that the circuit is numbered from back-end (output) to front-end (input) in sequence.

FIG. 4 shows that the circuit is numbered from front-end (input) to backend (output) in sequence.

FIG. 5 shows the (V, E) pairs of Fig. 4 numbered circuit display on the V-E plane.

FIG. 6 illustrates some special (V, E) distribution cases.

FIG. 7 shows a modified common multiplicative congruential random number series generator (ANSI C program).

FIG. 8A shows that interleave cutting the IC circuit can get an initial nearly max-cut partitioning status.

FIG. 8B separately randomizes the two part nodes in FIG. 8A.

FIG. 9 shows the ENISLE steps including a initialize step, and/or a randomize step, basic phase one sorting and different additional sorting phases.

FIGS. 10A-10F separately show an example of the detailed basic steps of the ENISLE.

FIGS. 11A-C shows an example of the steps of ENISLE.

FIG. 12 is a flow chart diagram according to the present invention type 2A.

FIG. 13 shows the usage of bit field structure, it effectively reduce the memory requirement.

FIG. 14 shows the usage of radix sorting.

FIGS. 15A-15F show an example wherein ENISLE effectively solved the mincut and ratio-mincut partitioning problem at the same time.

FIG. 16 shows another example solved by ENISLE.

FIGS. 17A-17C show some useful data display compression techniques.

FIGS. 18A-B shows the processes of the example in Fig. 15 by display compression representation in Fig. 17A.

FIG. 19 shows a non-uniformly distributed condition.

FIGS. 20A-20C shows the relationship between cut numbers and the initial (V, E) pairs distributed condition/entropy.

Detailed Description of the Preferred Embodiment

The present invention discloses a method of min-cut and ratio min-cut partitioning. To solve the problems mentioned above, a description of the preferred embodiments of this invention has diagrams shown in FIGS. 1-2. FIGS. 1 and 2 are diagrams explaining the theory of an embodiment of this invention. The principle of the present invention is to transform a circuit into a hyper-graph or netlist G=(V, E) which contains node and edge information. V indicates the set of nodes that represent components of the circuit and E is the set of hyperedge that represents the nets of the circuits. Each hyperedge or net connects two or more nodes. In general, the output of a node is connected to the inputs of a plurality of other nodes by a net.

In FIG. 1, the diagram indicates min-cut partitioning whereas FIG. 2 represents the ratio min-cut partitioning. When (V, E) pairs approach uniform distribution on the V-E plane, if the min-cut state is found, the "out-line area" from VE will approach VE/2, similar to vapor compression behavior.

FIG. 12 discloses the steps that comprise ENISLE the first step 12100 is to transform or map a circuit into a (V, E) plane, or initialize the V-E plane. The circuit information is then transformed into a hyper-graph or netlist G=(V E) which contains node and edge information. The V-E plane includes a matrix having column and row information. For example, the column includes the edge information, while the row includes the node information, which constructs the V-E pairs. It is appreciated that the parameters indicating columns and rows might be interchanged.

The Description of (V, E) pairs on the V-E plane is shown as follows, a circuit is used as an example rather than limiting the present invention:

The circuit in FIG. 3 is the same as the circuit in FIG. 4. It is composed of 2-input NOR gates, 2-input NAND gates and inverter gates. Every node (gate) connects 2~3 edges (pins). FIG. 3 shows the nodes and edges of the circuit numbered from back-end (output) to front-end (input) in sequence. On the contrary, FIG 4 shows the nodes and edges of the circuit numbered from front-end (input) to backend (output) in sequence. Because VLSI circuits are hyper-graph, compound tree-based structures, or like a forest, the circuits were also numbered by the depth first search (DFS) style or the Breadth First Search (BFS) style.

The numbered (V, E) pairs of FIG. 4 planeare shown in FIG. 5. Because the numbers of the circuit are in sequence, they are diagonally distributed on the V-E plane, like a narrow leaf. The width W of the “leaf” on the V-E plane, is in proportion to the “width” of the circuit. The width W times the levels of the circuit (tree), are approximately equal to the size (node) of the circuit.

In general, the larger the nodes, the larger the edges. Because this circuit is simple and small, one can view the V-E plane without any further processes, to find the bi-part cuts are 6 cuts, and the ratio tri-part cuts are 6 cuts. Although the min-cuts are 5 cuts, and ratio min-cuts are 4 cuts, the 6 cuts answer is good enough for industrial uses. If other min-cut methods are used, one may waste time doing meaningless minor improvements.

FIG. 6 shows some special (V E) distribution cases, namely, non-uniform distribution. It is often seen when describing the row-based placement in sequence that the "outline area" of (V E) pairs occupied on the V-E plane is far less than VE. This characteristic is unlike vapor compression behavior, which is conversely, like "melting" the material. Therefore, "heating" may be necessary to add "entropy" to it. Further, these cases ease to resolve the blocks and some small blocks will be automatically resolved by the method.

Common Quasi-Random Case

The example of ANSI C function rand in FIG. 7 uses a common low-end multiplicative congruential random number generator (ex. Borland C++ Compiler, 1992 DOS version) with period 232 to return successive pseudorandom numbers in the range from 0 to 2^{15}. This means it can handle a 2^{15} (32,768) node-size circuit under this compiler. After suitable modification, a quasi-random number series without repeated-present numbers is generated.

The Quasi-Random Case Under Nearly Max-Cut Reservation

Hypergraphs are systems of sets which are conceived as natural extensions of graphs, wherein elements correspond to nodes and sets correspond to edges, which are allowed to connect more than two nodes. Hypergraphs are typical compound tree-based structures. The same level nodes in tree-based structures do not connect each other. In other words, nodes on the same level do not have edges.

VLSI circuits are divided into two parts, one part mainly contains odd-level nodes, another part mainly contains even-level nodes. By this interleave cutting concept, due to the nodes between the same level have no edges, one gets a nearly max-cut partitioning of the VLSI circuit. An example is shown as FIG. 8A. Separately randoming the order of these two part nodes as shown in FIG. 8B, has several advantages: the initial stage not only has a higher entropy, but also holds the nearly max-cut reservation. This leads to higher converge speeds. The following sections are adopted by this improved method.

ENISLE:

EDGE NODE INTERLEAVE SORT FOR LEACHING AND ENVELOP

Step 12010 in FIG. 12, determines whether (V, E) pair distribution is uniform or not. If the distribution approaches non-uniform distribution, then randomize the (V, E) pairs on the V-E plane in step 12020.

On the contrary, if the distribution approaches uniform distribution, a phase one procedure or edge interleave I is performed for sequentially arranging allocations of the V-E pairs according to the magnitude of each node or edge, thereby obtaining min-cut or/and ratio min-cut partitioning. The phase one procedure includes the steps of:

(1) Performing a first sorting step (12030) from the edge view based on a first side, preferably, the bottom side of the V-E plane;

(2) Performing a second sorting step (12040) from the node view based on a second side, preferably, the right side of the V-E plane;

The aforementioned two steps may set the upper triangle area that contains almost no data therein.

(3) Performing a third sorting step (12050) from the edge view based on a third side, preferably, the top side of the V-E plane;

(4) Performing a fourth sorting step (12060) from the node view based on a fourth side, preferably, the left side of the V-E plane.

Similarly, the lower triangle area also contains almost no data therein. It is appreciated that almost all of the information is gathered from the diagonal line of the V-E matrix or the V-E plane.

The above phase one procedure includes four sorting steps, called edge interleave I step, ENEN. From this step, one may obtain the cutting configuration.

Step 12070 initializes the node set record. FIG. 9 shows the steps including phase one sorting and different additional sorting phases. As shown in the diagram, the additional sorting phase includes a plurality of sorting modes comprising recurring order type 2 A: N(R) E(T) N(L), type 213: N(R) E(T) N(L) N(R) E(B) N(L), type 2C: N(R) E(T) N(L) E(B) E(T) N(L) E(B) N(R) N(L) E(B) N(R) E(T) E(B) N(R) E(T) N(L), type 2D: E(B) N(R) E(T) N(L), and some other recurring order and clustering techniques. Each type returns similar results, an approach to a min-cut solution

FIG. 12 uses the type 2A as an example. It is appreciated that other types can be used. In step 12080, a fifth sorting step is used from the node view based on the second side, followed by performing a sixth sorting step 12090 from the edge view based on the first side/third side. Step 12100 determines whether the node set is still interchanged or not. If the node set is no longer interchanged, go to the end. Otherwise, a further sorting step is performed from the node view based on the fourth side, step 12110. In step 12120, it is determined whether the node set is still interchanged or not. If the node set is still interchanged, then step 12080 is performed again. Otherwise, an optimal min-cut or/and ratio min-cut partitioning is obtained.

The present embodiment can intuitively determine whether uniform distribution is present or not in the final diagram without needing any additional computing about correlation coefficients or co-variances. Suppose that the (V, E) pairs are not uniformly distributed on the V-E plane. If it is not randomized, the result may be a worse cut solution and the program leaves the loop (i.e., a non-determined/infinite loop does not occur).

FIGS. 10A-10F show an example of the phase one steps according to the present invention. It indicates that we may soon obtain the min-cut solution.

FIG. 10A illustrates an example of a circuit having 14 edges and 15 nodes. An initialize step is performed for mapping the circuit information into a V-E plane, as shown in FIG. 10B. The hyper-graph or netlist G=(V, E) contains the information of node and edge information. Each hyperedge or net connects two or more nodes, the relationship of the nodes are shown in the hyper-graph.

FIG. 10B illustrates the result after the first sorting from edge view based on the bottom side based on the bottom edge. It sequentially arranges the allocation of the V-E pairs according to the magnitude from high to low or low to high.

The sorted V-E plane may be presented as FIG. 10C from the node view based on the right side. Therefore, the sorting from the right side is illustrated on the right part in FIG. 10C, (i.e., the V-E pairs are arranged according to the magnitude of each node or edge based on the right side).

A sorting step from the node view based on the top side is performed and illustrated in FIG. 10D and the relocation result is shown on the right part in FIG. 10D.

Similarly, the last sorting step is illustrated in FIG. 10E, which is from the node view based on the left side. Next, the V-E pairs are arranged sequentially according to the magnitude or value from high to low. The result is shown on the right side of FIG. 10E. The optimal min-cut solution is shown in FIG. 10E. The V-E pairs may reconfigure to a min-cut or ratio min-cut partitioning circuit. The circuits may be seen in FIG. 10F. The total steps in FIG. 10A-F are summarized in FIGS. 11A-C.

In the ENISLE algorithm, a carefully arranged memory requirement is necessary. As shown in FIG. 13, the use of a bit field structure reduces the memory space to one eighth. If it has 100K nodes and 550K edges, the program will need about 6.4 GB virtual memory space. A powerful sorting engine decides the performance of the method needed to sort the very mass numbers. Using radix sort may handle this problem effectively, as shown in FIG 14. If the circuit is much larger, multi-level methods may be considered. Refer to D. E. Knuth, Sorting and Searching. Addison-Wesley, 1973 for radix sorting and for multi-level methods, one may refer to C. J. Alpert, J. H. Huang, and A. B. Kahng, “Multilevel circuit partitioning,” in Proc. Design Automation Conf., 1997, pp. 530-533.

FIGS. 15A-15F show an alternative example solved by ENISLE. This method is similar to the last embodiment. ENISLE effectively solves the min-cut and the ratio min-cut partitioning at the same time. FIG. 16 and 19 further show successful examples solved by ENISLE. In FIG. 19, the V-E pairs are not uniformly distributed on the V-E plane. If the randomizing step is not carried out, it performs the converge procedure. It shows that the optimal solution (2 cuts) cannot be obtained, but obtains a nearly optimal solution (3 cuts). It can be determined whether there is uniform distribution or not by the diagram.

(V, E) PAIRS DISPLAY AND REPRESENTATION

Originally display (V, E) pairs on a V-E plane

One can scroll the screen, like scrolling a spreadsheet, to observe the (V, E) pairs distributed condition.

DISPLAY DATA COMPRESSION

Display compression by different colors

For example, on a 1280 x 1024 pixels x 24 bits true color display monitor, assume 1280 x 16 bits edges / 1024 x 8 bits nodes = 20480 edges / 8192 nodes per screen, or 1024 x 24 bits edges / 1280 bits nodes = 24576 edges / 1280 nodes per screen.

Display compression by different light intensity and/or different patterns on a monochrome viewpoint

Some useful data compression technique examples are shown in FIGS. 17A-17C. L nodes x W edges (V, E) pairs rectangle region (if L = W, this is a square) compose a block. The more (V, E) pairs in the block, the higher light intensity. If no (V E) pairs are in the block, it is thick darkness.

Several other color quantities like hue, saturation, brightness, tints, tones, shades and luminance also can be adopted to represent the amount of the (V, E) pairs in a block.

Display compression may not show the exact position of the (V, E) pairs, it may only be displayed on a block, but a larger sized V-E plane or the whole V-E plane can be viewed on a monitor screen. The exact (V, E) pair positions will still be held, so one can zoom in the V-E plane to view detailed local (V, E) pair distribution, or zoom out to watch global (V, E) pair distribution. The processes of the example shown in FIGS. 15A-F and demonstrated by display compression in FIG. 17A are shown in FIGS. 18A-B.

The above methods can all be directly be observed at every iterative to: determine if there is improvement, get useful information or decide to manually halt the procedures or not, if necessary. This is suitable for IC industrial EDA for certain cut constraints under non-uniformly distributed cases.

A method for display data compression techniques by different light intensity and/or different patterns on a monochrome viewpoint, comprising:

displaying (V, E) pairs on an initial V-E plane shown on a monitor screen to observe the initial (V, E) pairs distributed condition, wherein the V indicates nodes that represent components of the circuit and wherein the E indicates edges that represents the nets of the circuits;

setting L nodes x W edges (V, E) pairs rectangle region to compose a block, wherein the L and W are integers;

defining the more (V, E) pairs in the block to be displayed by the relatively high light intensity to the less (V, E) pairs in the block; and

watching relatively large size of V-E plane or a whole V-E plane to the initial (V, E) plane on the monitor screen, wherein the exact (V, E) pairs positions still be held, thereby zooming in the V-E plane to watch detail local (V, E) pairs distributed condition, or zooming out to watch global (V, E) pairs distributed condition on the monitor screen.

The present invention provides a method for displaying data compression techniques by different light intensity and/or different patterns on a monochrome viewpoint, comprising:

displaying (V, E) pairs on an initial V-E plane shown on a monitor screen to observe the initial (V, E) pairs distributed condition, wherein the Vindicates nodes that represent components of the circuit and wherein the E indicates edges that represents the nets of the circuits;

setting L nodes x W edges (V, E) pairs rectangle region to compose a block, wherein the L and W are integers;

defining the less (V, E) pairs in the block to be displayed by the relatively high light intensity to the more (V, E) pairs in the block; and

watching relatively large size of V-E plane or a whole VE plane to the initial (V, E) plane on the monitor screen, wherein the exact (V, E) pairs positions still be held, thereby zooming in the V-E plane to watch detail local (V, E) pairs distributed condition, or zooming out to watch global (V E) pairs distributed condition on the monitor screen.

The alternative embodiment according to the display data compression techniques shown by different light intensity and/or different patterns on a monochrome viewpoint may define the lesser (V, E) pairs in the block compared to the relatively high light intensity of the greater (V, E) () pairs in the block.

Alternatively, the present invention provides a further method for displaying data compression techniques by different color and/or different patterns on a monochrome viewpoint, comprising:

displaying (V, E) pairs on an initial VE plane shown on a monitor screen to observe the initial (V, E) pairs distributed condition, wherein the V indicates nodes that represent components of the circuit and wherein the B indicates edges that represents the nets of the circuits;

setting L nodes x W edges (V, E) pairs rectangle region to compose a block, wherein the L and W are integers;

defining the more (V, E) pairs in the block to be displayed by the relatively bright color to the less (V, E) pairs in the block; and

watching relatively large size of V-E plane or a whole V-E plane to the initial (V, E) plane on the monitor screen, wherein the exact (V, E) pairs positions still be held, thereby zooming in the V-E plane to watch detail local (V, E) pairs distributed condition, or zooming out to watch global (V, E) pairs distributed condition on the monitor screen.

Alternatively, the above method for displaying data compression techniques may define the lesser (V, E) pairs in the block to be displayed by the relatively bright color of the greater (V, E) pairs in the block instead of the aforementioned definition.

As mentioned in FIG. 6, it shows (V, E) pairs non-uniformly distributed leading to the probability of getting the min-cut solution being relatively small. The work finds the relationship between cut numbers and the initial (V, E) pairs distributed condition/entropy is a very important issue. If the cuts of the higher initial (V, E) pairs distributed potential/entropy approach to max-cuts, or under nearly max-cut reservation, the higher the probability of getting a min-cut. FIG. 20A, FIG. 20B and FIG. 20C show the relationship between cut number and initial (V, E) pairs distributed condition/entropy. The cut number j>k>min-cut, k is the second optimal cut and the j is the third optimal cut. The higher the initial potential, the greater the probability of achieving the target min-cut.

Since ENISLE is different than any other min-cut partitioning method, ENISLE does not improve or modify other existing min-cut partitioning methods. Therefore, the present invention does not concentrate on the comparison with them, the main focus is on the demonstration of the new method.Vertex (node) min-cut also can be implemented by the proposed method, only add a transformation step:

$$G = (V, E) \rightarrow (E, V) = (V', E') = F$$

The work can get the minimal edge cuts of the network netlists F, and these are the minimal node cuts of the original network G. It may be useful on network flow problems. The present invention indicates that one can effectively solve min-cut partitioning and the ratio min-cut partition at the same time by global viewpoints. The ENISLE method not only uses node information but also uses edge information. Hundreds of netlists experiments have been processed and found the importance of (V, E) pair distributed condition. If (V, E) pairs approach uniform distribution on the V-E plane, one can soon get the optimal solution with no NP problem. If there is non-uniform distribution, or requires certain cut constraints (non-min-cut), the ENISLE method can provide an intuitive heuristic nearly optimal solution which is very suitable for IC industrial EDA usage.

As is understood by a person skilled in the art, the foregoing preferred embodiments of the present invention are illustrations of the present invention rather than limitations of the present invention. The embodiments are intended to cover various modifications and similar arrangements included within the spirit and scope of the appended claims, the scope of which should be accorded the broadest interpretation so as to encompass all such modifications and similar structure. Thus, while the preferred embodiment of the invention has been illustrated and described, it will be appreciated that various changes can be made therein without departing from the spirit and scope of the invention.